REMARKS

This application has been reviewed in light of the Office Action dated November 25, 2003. Claims 1-37, 40, 42-52, 54, 56, 57, 59-74, 76, 78-82, 85, 87-89, 91, 93, 94, 96-110, and 113-116 are presented for examination, of which Claims 1, 11, 20, 26, 31-33, 51, 52, 65, 73, 74, 78-80, 96, 97, 108, and 113-116 are in independent form. Claims 38, 83, 84, 92, 95, 111, and 112 have been canceled, without prejudice or disclaimer of their subject matter, and will not be mentioned further. Claims 1-3, 10-12, 19-22, 25-27, 30-35, 37, 40, 42, 45, 49-52, 56, 57, 59-67, 69, 71-74, 78-80, 82, 85, 87-89, 91, 93, 94, 96, 97, 99-101, 103-105, 108, 110, and 113-116 have been amended to define more clearly what Applicants regard as their invention. Favorable reconsideration is requested.

Claim 96 was rejected under 35 U.S.C. § 112, second paragraph, as indefinite.

Claim 96 have been carefully reviewed and amended as deemed necessary to ensure that it conforms fully to the requirements of Section 112, second paragraph, with special attention to the point raised on page 2 of the Office Action. Specifically, Claim 96 has been amended such that the phrase "the first hierarchical structure" has been amended to read --the hierarchical structure--. It is believed that the rejection under Section 112, second paragraph, has been obviated, and its withdrawal is therefore respectfully requested.

Claims 1-37, 40, 42-52, 54, 56, 57, 59-74, 76, 78-82, 85, 87-89, 91, 93, 94, 96-110, and 113-116 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,191,797 (*Politis '797*).

As shown above, Applicants have amended independent Claims 1, 11, 20, 26, 31-33, 51, 52, 65, 73, 74, 78-80, 96, 97, 108, and 113-116 in terms that more clearly define the present invention. Applicants submit that these amended independent claims, together with the

remaining claims dependent thereon, are patentably distinct from the cited prior art for at least the following reasons.

The aspect of the present invention set forth in Claim 33 is a method for optimizing an expression tree. The expression tree represents a compositing expression for compositing an image and comprising a plurality of nodes, where each node of the expression tree represents an object of the image or an operation for combining sub-expressions of the compositing expression. The method includes determining an opacity region representation for at least one node of the expression tree, the opacity region representation being assigned one or more of three predetermined values. Each predetermined value distinctly identifies whether a corresponding sub-region is an opaque region, a transparent region or a partially transparent region. The method also includes optimizing the expression tree by determining an obscurance region representation for at least the node of the expression tree base on an analysis of the opacity region representation associated with the node of the expression tree. The obscurance region representation being assigned one or more of a plurality of further predetermined values, where each further predetermined value distinctly identifies whether a corresponding sub-region is visible in the image.

A notable feature of Claim 33 is determining an opacity region representation for at least one node of the expression tree, where the opacity region representation being assigned one or more of three predetermined values and each predetermined value distinctly identifies whether a corresponding sub-region is an opaque region, a transparent region or a partially transparent region. Another notable feature of Claim 33 is optimizing the expression tree by determining an obscurance region representation for at least the node of the expression tree base on an analysis of the opacity region representation associated with the node of the expression tree, the obscurance region representation being assigned one or more of a plurality of

further predetermined values, where each further predetermined value distinctly identifies whether a corresponding sub-region is visible in the image.

As described at page 17, lines 10-18, of the present specification, for obscurance analysis, the homogeneous regions of an image are of interest, that is, those regions corresponding to parts of objects that are of different levels of opaqueness (i.e. fully opaque, fully transparent or partially transparent regions). These regions are needed in order to take full advantage of possible optimization opportunities that exist in the presence of the various operations (i.e., OVER, IN, OUT, ATOP, etc. . . .). Operators such as OVER require the fully opaque regions to determine where one object follows another, whereas operators such as IN and OUT require the fully transparent regions to determine how an object clips out parts of other objects.[1]

As further described at page 17, lines 25-28, of the present specification, each leaf node of a quadtree is assigned a value representing the colors black, white, and grey respectively, depending on whether a corresponding region in the image space is fully opaque, fully transparent or partially transparent, respectively. As described at page 20, lines 23-29, of the present specification, obscurance analysis requires knowledge of the opacity information of each object, in the form of opacity quadtrees, so that regions where objects have been hidden or clipped out by other regions can be identified. These obscured regions are generally irregular in shape and can also be represented using quadtrees, referred to as obscurant quadtrees. Unlike opacity quadtrees, obscurance quadtrees preferably contain only two distinct node values instead of three. The two distinct node values being '1' where the object is hidden, and '0' where the object is visible.

1/It is to be understood, of course, that the claim scope is not to be limited by the details of the described embodiments, which are referred to only to facilitate explanation.

Obscurance quadtrees are computed from the opacity quadtrees, as described at page 21, lines 12-22, of the present specification. An obscurance quadtree represents the union of all obscured regions represented by a corresponding leaf node. As each node in a compositing tree inherits the obscured regions of the nodes parent node, the obscurance quadtrees are propagated in a downwards tree traversal. The process concludes with a final obscurance quadtree which can be used to limit the amount of processing required to render the graphics object corresponding to the particular leaf node.

In contrast, *Politis '797* discusses a method of optimizing an expression tree for compositing an image where the region represented by a node is compared to a quadtree corresponding to a region represented by another node. As disclosed, at column 5, lines 25-28, a <u>single</u> quadtree corresponding to unobscured portions of graphical elements is returned from a node for possible further processing at other nodes. In Table 2 of *Politis '797*, set operations such as intersection, union and subtraction are used to compute this single quadtree returned from a node depending on the compositing operator at the node. For example, for the over operator, Table 2 shows that the returned quadtree is $qL \cup qR$, being the union between quadtrees qL and qR. These set operations imply that the quadtrees employed by *Politis '797* are strictly binary region representations, in that each quadtree comprises nodes representing either sub-regions belonging to the represented region (so called '1' or 'ON' nodes) or sub-regions not belonging to the represented region (so called '0' or 'OFF' nodes). More specifically, in the quadtrees employed by *Politis '797*, the '1' nodes represent fully opaque sub-regions of one or more objects that can potentially obscure other objects, while the '0' nodes represent all remaining non-fully opaque sub-regions of the one or more objects, which include and do not distinguish between sub-regions that are fully transparent and sub-regions that are partially transparent. Thus, nothing has been found in *Politis '797* that would teach or even suggest using

region representation that can distinctly identify sub-regions that are opaque, transparent or partially transparent, as recited in Claim 33.

In the example of column 15 of *Politis '797*, a quadtree is created representing a region of an image occupied by a circle. The quadtree is passed back to the parent node of the node representing the circle and is stored as the obscuring region of the node representing the circle. This implies that the quadtree identifies sub-regions of that circle that are fully opaque (since partially transparent and transparent regions can not obscure other objects. If the circle contains sub-regions that are partially transparent, then they are excluded from the region represented by the quadtree. The intersection of this quadtree and another quadtree representing the unobscured fully opaque portions (i.e., those portions which have the potential to obscure other graphical elements) of the other child node of the parent node is then determined. The result of the intersection is a quadtree representing a fully opaque region of the parent node which can obscure other graphical elements of the corresponding expression tree. However, nothing has been found in *Politis '797* that would teach or suggest determining an opacity region representation for a node of the expression tree and then determining <u>an obscurance region representation</u> for the node of the expression tree based on an analysis of the opacity region representation associated with the node of the expression tree, as recited in Claim 33.

Column 3, line 61, to column 4, line 63, of *Politis '797* discusses the use of alpha values used in compositing operations. *Politis '797* discloses a value Do representing the resultant alpha channel value of a compositing operation, an alpha value Ao corresponding to a pixel having a color Ac and an alpha value Bo corresponding to a pixel having a color Be. The values Ao and Bo are used to determine the value Do representing the resultant alpha channel value. These alpha channel values are not directly used in determining quadtree region representations of objects, nor are they directly used in optimizing the expression tree. Instead

the formulae listed in Table 1 for computing Do for the different compositing operators are used to identify the conditions under which fully opaque pixels are certain to be obtained from a compositing operation (i.e., when Do = 1), which in turn determines what quadtree gets passed to each operand of the operator and how a return quadtree is computed during expression tree optimization, as specified in Table 2. Thus, as dicussed in *Politis '797*, alpha values do not play any role in distinctly identifying, within a region representation of a node of the expression tree, sub-regions that are fully opaque, fully transparent or partially transparent. Further, the compositing operations for combining two portions of a single image of *Politis '797* does not teach or suggest distinctly identifying whether a corresponding region represented by a node of the expression tree is an opaque region, a transparent region or a partially transparent region, as recited in Claim 33

Accordingly, Applicants submit that *Politis '797* does not teach nor even suggest the particular claimed limitations of Claim 33 of determining an opacity region representation for at least one node of the expression tree, where the opacity region representation being assigned one or more of three predetermined values, each predetermined value distinctly identifying whether a corresponding region represented by the node of the expression tree is an opaque region, a transparent region or a partially transparent region. Further, *Politis '797* does not teach or suggest the particularly claimed limitation of optimizing the expression tree by determining an obscurance region representation for at least the node of the expression tree based on an analysis of the opacity region representation associated with the node of the expression tree, the obscurance region representation being assigned one or more of a plurality of further predetermined values, each further predetermined value distinctly identifying whether a corresponding region represented by the node of the expression tree is visible in the image.

For at least the above reasons, Applicants submit that Claim 33 is clearly patentable over *Politis '797*.

Independent Claims 1, 11, 20, 26, 31, 32, 51, 52, 65, 73, 74, 78-80, 96, 97, 108, and 113-116 include substantially similar features as those discussed above in connection with Claim 33. Accordingly, Claims 1, 11, 20, 26, 31, 32, 51, 52, 65, 73, 74, 78-80, 96, 97, 108, and 113-116 are believed to be patentable for at least the same reasons as discussed above in connection with Claim 33.

A review of the other art of record has failed to reveal anything which, in the Applicants opinion would remedy the deficiencies of the art discussed above as references against the independent claims of the present application. Those claims are therefore believed patentable over the art of record.

The other rejected claims in this application depend from one or another of the independent claims discussed above, and, therefore, are submitted to be patentable for at least the same reasons. Since each dependent claim is also deemed to define an additional aspect of the invention, individual reconsideration of the patentability of each claim on its own merits is respectfully requested.

This Amendment After Final Action is believed clearly to place this application in condition for allowance and, therefore, its entry is believed proper under 37 C.F.R. § 1.116. Accordingly, entry of this Amendment After Final Action, as an earnest effort to advance prosecution and reduce the number of issues, is respectfully requested. Should the Examiner believe that issues remain outstanding, it is respectfully requested that the Examiner contact Applicants' undersigned attorney in an effort to resolve such issues and advance the case to issue.

In view of the foregoing amendments and remarks, Applicants respectfully requests favorable reconsideration and early passage to issue of the present application.

Applicants' undersigned attorney may be reached in our New York office by telephone at (212) 218-2100. All correspondence should continue to be directed to our below listed address.

Respectfully submitted,

Attorney for Applicants

Registration No. 26,718

FITZPATRICK, CELLA, HARPER & SCINTO
30 Rockefeller Plaza
New York, New York 10112-3801
Facsimile: (212) 218-2200
NYMAIN404780